AD

AD-E402 907

Technical Report ARFSD-TR-00001

A STUDY OF AUTONOMOUS MICRO-ROBOTS AND THEIR APPLICATION TO COMPLEX ENVIRONMENTS VOLUME I

Christopher Niles
Thehue Tran

May 2000

COMMITTED TO EXCELLENCE
US ARMY
TANK AUTOMOTIVE AND
ARMAMENTS COMMAND
ARMAMENT RDE CENTER

U.S. ARMY ARMAMENT RESEARCH, DEVELOPMENT AND ENGINEERING CENTER

Fire Support Armaments Center

Picatinny Arsenal, New Jersey

REPORT DOCUMENTATION PAGE					*Form Approved* *OMB No. 0704-01-0188*

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (*From - To*)
May 2000	Final	

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
A STUDY OF AUTONOMOUS MCRO-ROBOTS AND THEIR APPLICATION TO COMPLEX ENVIRONMENTS, VOLLUME I	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHORS	5d. PROJECT NUMBER
Chistopher Niles and Thehue Tran	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
ARDEC, FSAC Fire Control & Life Cycle Software Engineering Division (AMSTA-AR-FSF) Picatinny Arsenal, NJ 07806-5000	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) TACOM/ARDEC
ARDEC, WECAC Information Research Center (AMSTA-AA-WEL-T) Picatinny Arsenal, NJ 07806-5000	11. SPONSOR/MONITOR'S REPORT NUMBER(S) Technical Report ARFSD-TR-00001

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report identifies and surveys the state of the art in enabling technologies supporting the major components of autonomous mobile robot design. The object of this effort is to determine to what extent ongoing industry and university programs can support the development of "lobster-sized," land based autonomous robots capable of traversing terrain consisting of vertical obstacles, canyons, and diverse surface textures in all-weather conditions. Particular attention is placed on path planning with limited apriori knowledge and primitive sensors. Ideas on how to conduct 3D path planning of complex outdoor environments are discussed. Following the evaluation and survey, the report recommends areas requiring further research.

15. SUBJECT TERMS

Autonomous robot(s) Path planning system Micro-robot(s) Algorithms Robotic transportatin Complex work space

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (Include area code)
U	U	U	SAR	31	

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

CONTENTS

	Page
Introduction	1
Background	1
Robot Design	1
Simulation and Analysis	2
Terrain Mapping	3
Specifications for an Autonomous Robot in an Unknown Complex 3D Environment	4
Technical Difficulties in Building Autonomous Robotics Unit	4
Concerns for the Path Planning System	5
Current Available Technology	6
Various Sensor Systems Used for Path Planning Systems on Autonomous Robots	6
Forms of Locomotion	8
Commonly Used Algorithms	8
Computer Aided Design Programs	11
Projects and Research	13
Discussion	22
Conclusions	24
References	25
Distribution List	27

FIGURES

		Page
1	Path generated using A*	10
2	Path generated using D*	11
3	Ariel robot	14
4	Artist view of Robug III	15
5	Troody, one of MIT’s walking biomimetic robots	16
6	Millibot shown with a quarter	17
7	CyberRAVE robots used to interact with software	17
8	NASA’s serpentine robot	19
9	Mama Loon with Baby Loon on back	20
10	The Walleye	21

INTRODUCTION

The conception for this report is based on an idea to have a small mobile robot crossing a complex terrain such as a bridge or obstacle-filled area. The robot would be able to conduct itself over or around any obstacles it may face. The ultimate goal is to determine how much further technology needs to advance to able to produce a completely autonomous mobile micro-robot capable of navigating itself through impressive difficulties. This type of robot would be extremely useful for exploration, inspection, or dangerous assignments. A volcanic eruption would pose too great of a risk for a human geological party to collect data whereas a small adaptable robot will be able to do the job without life endangerment. Another good example is bridge building inspections. With a small autonomous robot capable of examining the bridge, it would reduce costs that would have been allocated to rigging and traffic control needed to maintain the safety of a human crew. This report is a preliminary study into the maturity of the fields of robotic technology and its ability to meet the requirements of such a project. It will serve to identify areas of increased research and will give an overview of the many technical challenges associated with developing a small highly mobile robot to function in a complex three-dimensional environment. It will also present the many technologies available and in development, along with their advantages and disadvantages, that may aid in achieving the goal.

BACKGROUND

The primary fields considered for research include robotic design, path planning, simulation, analysis, and terrain mapping. A more detailed synopsis of the different technologies available is discussed later in this report.

All the background information was acquired through the use of materials made available by several distinguished companies and universities in the field of Micro-robotics and autonomous navigational research on their web sites. Several professors and engineers were contacted to discuss their research and projects as well as their expert opinions on different subjects of interest.

Robot Design

The first topic of research is searching different approaches of accomplishing robot design. A well-established part of robot design is the building of a prototype. Built for real-world testing purposes, a prototype performs its programmed tasks in an environment similar to what its operational environment will be. For example, a mobile robot crosses different terrain to test its stability, maneuverability, and durability. Prototypes accurately show how a design will act in operation, but they tend to be expensive and are not easily modified. Less costly alternatives are available.

The most common approach is Computer Aided Design (CAD). A CAD system takes all of the design work historically done by hand to the computer and allows highly accurate sizing and placing of all components in a design. An excellent feature of CAD is that the design can always be put together using its full size dimensions; scaling to fit a drawing on a page is no longer a concern for the draftsman. A second nice feature of CAD is its versatility. Modification of a design is clean and quick, and comparison of different modifications can be laid over the original design.

Most CAD today is accomplished in a full three dimensional environment. Developers are allowed to fully realize a design before a prototype is built. Accidental collision between components in motion is detected early in development. In addition, the manufactureability and service ability of a design are determined early in the process.

Computer Aided Design can help designers take their completed computer model to the next step of robot design. Computer Aided Design file formats are compatible with most simulation software. This allows designers to output the CAD model into a simulation program for virtual testing. The design does not need to be modeled in the simulation environment. The software automatically interprets the information from the CAD file.

Simulation and Analysis

Computer simulation is becoming an increasingly important part of robot design. It is very common to find simulation programs that will run a design through various motions to see how that design will react. This is most popular in automobile design. Traditionally, fully functional cars and trucks are put through rigorous road and collision tests for purposes such as vehicle safety and stability. Prototypes of cars are driven into walls and other cars to see what damages the vehicles incur. Several points of each vehicle are analyzed for movement and damage.

Data from each computer simulation is stored in memory so that it may be analyzed. The computer, using the given parameters and the depicted motions in the simulation, calculates any of a number of results. Typical calculations include the stresses at different points, the chance of slippage between the terrain and the mode of locomotion, and even the temperature of internal electronic components.

Advantages of Computer Simulation and Analysis

Time. The time required to set up an individual test is greatly reduced. Once the design is complete and imported into the simulation software, only the locations that will be analyzed need to be chosen. The computer will keep track of those locations throughout the test; no calibration of sensors is required.

Cost. The costs involved in testing are greatly reduced because less hardware needs to be built for testing purposes. The dynamics are programmed into the computer, as well as typical outside influences such as temperature and wind in the environment. Material characteristics of all surfaces on the vehicle and the landscape are also included into the simulation.

Space. Space no longer needs to be procured for a test. The only space required is for the computer which is already part of the lab.

Modification. Many tests can be run in quick succession, allowing a designer to quickly optimize his design. Parts can be moved and resized without having to re-drill holes and cut components to fit properly.

Support. Advanced simulation software is capable of aiding the designer in programming the final behavior of a robot. The designer can instruct the robot to perform in a certain way; and the program will write the commands in a form that the processor of the robot will understand.

Disadvantages of Computer Simulation

The software must be coded properly. It must be written in a way so that the computer can understand its instructions so the motions can be modeled accurately.

Material characteristics of the different components need to be included in the software. Therefore, initial development of a simulation program can be very complicated and require many programmers with practical knowledge in physics and dynamics to model the simulation world properly.

Although computer simulation models of real-world environments are very precise, minute outside influences are excluded from software packages. This is mainly due to limitations in computing power. Even though the influences may not cause much of a disturbance, they can cause some erratic behavior that does not materialize in the simulation. Only a prototype tested in the real world can demonstrate such behaviors. Therefore, it will be impossible to completely eliminate prototypes. They will always be a necessary final step in analysis.

Terrain Mapping

In the world of computer simulation, work is done in either robot design or terrain mapping. It is very difficult to find the two areas combined into a single software package. For the most part, computer programs commercially available either deal specifically with environmental modeling or robot design and analysis. Development of software that can handle both fields of design would aid designers in developing robots with the ability to travel terrain of higher complexity.

As stated earlier, robot simulation programs are useful to model movements of the components of a design and have limited capabilities in modeling robots crossing terrain. Most generated terrain is simple in its layout. Either it is drawn as a flat surface with obstacles for the robot to avoid or it is composed of simple hills and valleys to test the control behavior of an autonomous robot.

On the other side of computer design is terrain mapping. These programs perform functions such as landscape design, ground layer analysis, topography, erosion modeling, and water flow control. Very detailed geographic maps are created and usually combined with geographical data to create a Geographic Information System (GIS). Geographic Information System is a globally linked database whose information is displayed geographically. Using the earth as an image map, it can show average temperature, rainfall, population, etc. It is the most widely used global information database today. It is essentially used as a very large database for census information and geographically localized data. However, it is not useful for creating maps with which a computer designed vehicle could interact.

Industrial engineering employs some methods of terrain mapping. One method for characterization is statistical analysis. This is used especially when designing a new layout for an assembly line or a manufacturing shop. Statistical analysis starts by looking at the big picture. An engineer will examine all the different aspects of the process to be done at the shop. They look at the processes of the work, features of the building, location and length of travel between stations, possible delays of operation, ease of operation, ease of parts repair in case of maintenance or breakdowns, and assembly or manufacturing time for the entire process. The next step is to create a chart to

show how much time or effort is spent on each part of the work. Using the chart, they can then optimize their layout by doing things such as minimizing the difficult tasks in favor of simpler tasks and placing large machines in areas where high maintenance components are easily accessible. They look at ways of reducing the overall workload and time for the process.

SPECIFICATIONS FOR AN AUTONOMOUS ROBOT IN AN UNKNOWN COMPLEX 3D ENVIRONMENT

Technical Difficulties in Building Autonomous Robotics Unit

Before building an autonomous robotics unit, there are six characteristics that must be considered. By analyzing each characteristic and their effects, a useful guide will be made available for engineers to proceed in determining the proper various types of components needed.

Mission

The mission must first be fully determined before any other considerations can be evaluated. The mission may practically determine the other characteristics in building the autonomous robotic unit by itself. The detailing of the mission can be broken down into these classifications:

- Goal(s): the recognition of the goal location and the subsequent actions regarding that location
- Environment: the workspace that has a fixed boundary and includes the goals
- Obstacles: features found in the environment that may impede a robot's mobility
- Intangibles: possible constraints such as size or weight requirements

Size

The size of the autonomous robotic unit is an important characteristic. The mission may require a specific size constraint on the autonomous robotics unit for it to be able to achieve its goal. Therefore, the size is an important factor in building an autonomous robotics unit.

Locomotion

The type of locomotion system used will have a great impact on the capabilities of the robot. Various types of locomotion modes were created for robotics units, allowing them to overcome obstacles of different difficulties. The type of locomotion used will also affect the amount of power it consumes to achieve its goals. Therefore, a great deal of thought and consideration must be exercised before choosing a proper type of locomotion unit to ensure that it is capable of meeting its objectives in an accepted worst case scenario.

Power

The power unit must contain enough energy to allow the autonomous robotics unit to accomplish its mission. Important factors in deciding a suitable power unit would be its power, size, and weight based on any particular constraints found in the mission.

Performance

Various companies and universities have researched for new technologies that would increase and maximize the productivity and performance of their autonomous robotics unit. These may include sensors, provided mapping ability, and so forth. A list will be tabulated later in the report to describe these various abilities that may complement its performance.

Path Planning System

The path planning system is the approach and behavior that the autonomous robotics unit takes to complete its mission. Several path planning systems exist currently either in the market or in development. There are several concerns that have to be determined in order to find a suitable path planning system for our needs.

Concerns for the Path Planning system

There are five criteria to consider when building a path planning system.

Equipment

Various devices such as sensors may be accessible to the autonomous robot that may allow it to perform more efficiently. The path planning system must integrate these devices into the program to ensure that it will fully utilize them for maximum potential returns.

Obstacles

The path planning system must be capable of understanding and acknowledging all obstacles found in the environment. It must be able to calculate its sizes and irregularities so that it may be able to determine various ways to either overcome or avoid them. It must also have the ability to distinguish a goal from other obstacles.

Errors

Working in a complex 3D obstacle-filled terrain, errors are sure to be compiled within the path planning system. These errors could be comprised from a various lot, whether forced or unforced. Examples include the autonomous robot falling to its sides, a change in the environment, or failure in detecting movement due to slippage of the wheels. A combination of various sensors and position systems may reduce these errors to an acceptable level.

Recognizing Patterns

Recognizing patterns and map location in the landscape will allow the path planning system to better understand its environment. By being able to recognize patterns and map location, it will prevent the autonomous robot from making a complete circle in search and create a whole new map for an area it has already covered.

Decision Making Process/Algorithms

The path planning system will be required to calculate various possible paths. However, the decision should not be as simple as choosing the shortest route. Sometimes, it would be better for the autonomous robot to choose the most power-efficient route or the easiest route. By providing good algorithmic programming, a wise methodical decision making process can be constructed to maximize efficiency and success rate.

CURRENT AVAILABLE TECHNOLOGY

Various Sensor Systems Used for Path Planning Systems on Autonomous Robots

Wheel Encoder

The majority of the current robots update their location by integrating data from their wheel encoder, which count the number of wheel rotations that are converted on a 2-polar map grid. This system of tracking position is extremely basic and old compared to current technologies available today. It works well on smooth unobstructed surfaces but fares poorly on obstacle filled surfaces. If the robot slips, the wheels do not rotate properly and the robot encoder will not register the change in position. Also if the robot gets stuck but the wheels are able to continuously rotate, the encoder will register false calculation steps to its 2-polar map grid.

Dead Reckoning System

Dead Reckoning is an upgraded position system based on the wheel encoder. It uses various sensors to calculate time, distance, velocity, and polar coordinates for better positional results. However, the accuracy of this system is entirely dependent on the measurement tools, which are often relatively crude. It is still prone to reporting erroneous data due to outside influences on the robot such as slippage, albeit to a lesser degree. Extremely useful if combined with other systems for a double positional check (ref. 1)

Global Positioning System

A worldwide digital-navigational system composed of a constellation of 24 satellites and their ground stations. The basis of how GPS works is to use a receiver to collect measurements from available satellites as reference points via travel time of digital signals to triangulate the current location of the receiver. Extremely useful, but its technology is still not advanced enough to work efficiently at an affordable cost. None of the commercially available GPS systems are capable of working inside buildings nor provide the necessary resolutions.

Available GPS:

- Low-cost, single-receiver single project funding (SPS) projects (100-m accuracy)
- Medium-cost, differential SPS code Positioning (1 to 10-m accuracy)
- High-cost, single-receiver PPS projects (20-m accuracy)
- High-cost, differential carrier phase surveys (1-mm to 1-cm accuracy)

Landmark System

A system designed to locate itself by using deployed landmarks in an environment to triangulate its position. The accuracy, if this system decreases as the area the landmarks cover increases. This system suffers due to its dependency on landmarks to work.

Voronoi Graphing System

A geometrical approach in locating an autonomous robot in an area. The system uses sonar sensors to locate its position by matching up nodes and edges, calculating the adjacency relationship between them. A complementary capability of this system is the robot's ability to determine its location in a partially explored map or to ascertain that it has entered a new territory. This system uses a generalized Voronoi graph (GVG), which is a map embedded in the robot's free space that captures the topologically salient features of the free space. The robot is then able to propagate the coordinates of each point on the GVG from the known location of one point, which is usually the start point, specified as point (0,0,0).

Most approaches are constantly trying to update the robot's coordinates, relative to a global frame, whereas the approach of this system locates the robot on a map and never updates the robot's location. The system recognizes and labels particular areas by matching up nodes and the adjacency relationship between them. The robot essentially traces double equidistance until a sensor threshold is met, at which point the robot follows the obstacle boundaries. Thus, the system is capable of easily self-determining distinct places from sensor data. Distinct places are a subset of the nodes of the GVG, which is the set of points equidistant to three obstacles. Localization is achieved by matching these distinct places of the graph.

The incremental construction of the GVG has four key components:

- Explicitly "trace" the GVG edges
- Determine the location of the meet points (GVG vertices)
- Explore the branches emanating from the meet points
- Determine when to terminate the tracing procedure

The robot traces a GVG edge until it detects a meet point, boundary point or Voronoi vertices. The meet point is where GVG edges meet. A geometrical calculation will then be used to find the precise location of the meet point. The stability of the resulting system will allow a conclusion that the robot will converge to the location of the actual meet point. Once at the meet point, the robot will determine the directions of the other GVG edges that emanate from it. It will then choose one of these GVG edges, pending on programming priorities, and follow it until it finds a new meet point with new GVG edges. This process will then build itself a whole map of the environment (refs. 2 and 3).

Forms of Locomotion

There are three types of locomotion currently available. Each type has advantages and disadvantages compared to the others. The locomotion used will dictate its ability to cross different types of terrain and obstacles of various difficulties.

Wheels

Types of wheels vary from numbers to arrangement. It draws the least amount of energy compared to the other modes of transportation and the most commonly used type of locomotion for robots. The algorithmic programming is also relatively easy to program. The reach it has is very limited however, and wheels cannot overcome obstacles of great difficulties.

Legs

Types of legs varying in shape and sizes have been created and programmed to walk, run, and even hop. It draws a moderate amount of energy compared to the other modes of transportation. If programmed correctly, a leg-type transportation mode can overcome practically any obstacles that are reasonably attainable. Unfortunately they are extremely difficult to build and to control in terms of balance, especially if they need to perform hard movements such as climbing. Walking and running robots seem to be the most preferred type way of moving. There are several jumping robots, but all of them require a lot of stabilization components, making them large and difficult to build smaller. Also, most hopping robots are attached to an external stabilizing bar to prevent them from falling to their side because they are not capable of maintaining their balance for a long period of time.

Serpentine

A relatively new type of locomotion system compared to the other modes. If designed correctly, serpentine robots can be very stable and are capable of traversing through any type of obstacles. The programming for a serpentine motion robot is much harder to code. The actual design and building of a serpentine robot is complex and difficult as compared to robots with wheels or legs. Another problem for serpentine robots is the costly energy consumption level they need to perform. All the current serpentine models use external power supply and computing.

Commonly Used Algorithms

Exploration Methods

Wall Hugging. The simplest of the maze solving algorithms is wall hugging. As the name implies, a robot will begin its search for the goal by following either the right or the left wall of the maze. For example, if it were to follow the left wall, it will turn left at every left turn of the maze. The robot remembers all of its movements and continues in this fashion until it reaches either a dead end or its goal. If a dead end is encountered, the robot backtracks until it finds a left turn that it has not tried previously. The robot will continue in this way until it either finds the goals or realizes that a solution is not possible.

Wall hugging is only useful in a two dimensional maze application. It cannot be used to find a goal in the middle of an environment if there are no walls leading directly to that goal. It is very simple to program because a map of the area is not necessary, only the start and finish points are required (ref. 4).

Flood-Fill. The flood-fill algorithm is simple to implement, but it requires that the robot know the layout of the maze before it begins. It was dubbed “flood fill” because it works as if all the passages in the maze are instantly filled with water. Any sections in which the “water flow” is stopped by a wall, or two flows collide in a loop, are cut off and become walls from the point of collision to the nearest junction. This happens until only the solution path is left (ref. 4).

Breadth-First. The breadth-first algorithm is more applicable for finding the solution to a computer-generated maze rather than one being solved by a robot. This is because the algorithm searches the entire maze before deciding on its first move. The algorithm is structured to begin its search of the maze at the highest level (the start point) and checks every cell at the level. It searches for the next best move to the next lower level of the maze. It continues through the maze until it either finds a solution or finds that it cannot continue to the next level. No matter what the complexity of the solution is, it will always take the same amount of time to find (refs. 4 and 5).

Depth-First. The depth-first algorithm is very similar to the breadth-first in that it is more applicable to computer generated mazes. Again the start and finish locations must be known before the algorithm is run. Unlike the breadth-first approach, the depth-first algorithm goes as deep into a possible solution path as it possibly can. If a dead end is found, the algorithm backtracks to the next available junction. This is generally a better algorithm than the breadth-first because of the possibility that all cells may not need to be explored. One drawback of this algorithm is that it may be temporarily delayed by looking for a local optimization and may not discover the global solution until much later (refs. 4 and 5).

Planning Methods

A*. A* is an algorithm that is guaranteed to find the shortest path from start to finish in a static two dimensional environment, given that there is a solution. As shown in figure 1, the area in question must be known and a map provided in order for the algorithm to function. As a planning method, the algorithm maps all moves before the robot begins its journey.

The A* algorithm operates by knowing the “state” of the robot, and not just the location. A state represents the position and direction of the robot. A* also uses a heuristic function to estimate the length from one state to another. Unlike exploration methods, A* will look at several possible moves along different possible solutions all at once and will sort them based on their cost. For example, the move with the lowest cost will be at the top of the list. The algorithm may work on one path for a little bit, then jump to another if the associated cost is then lower than if continuing the current path (refs. 4 and 5).

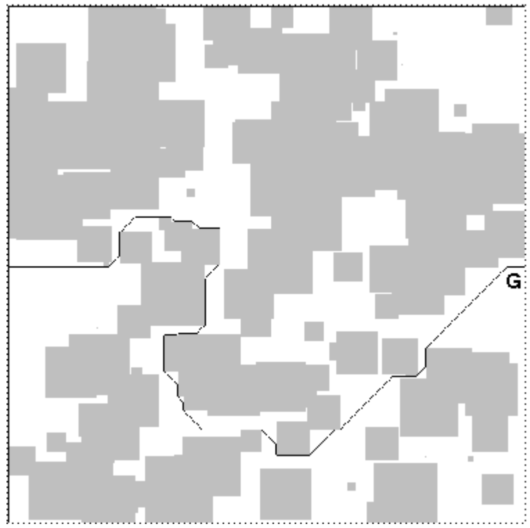


Figure 1
Path generated using A*

D*. D*, developed by a group headed by Anthony Stentz at the Robotics Institute at Carnegie Mellon University, is an algorithm that is similar to A* except that D* does not require that the environment be known or even static. Only the start and finish locations are known. The D* algorithm functions as both exploration and planning methods. First, an optimal path is planned using whatever information (if any) is available. Then the robot will move along that path until it encounters an obstacle that is not already on its map. It will then replan a small portion of its path towards its goal. The algorithm will continue to replan parts of the route of the robot as new obstacles come into view. Adjusting only a piece of its planned route and not the entire route at one time allows decisions to be made quicker and less memory to be used by the processor. The new obstacles are saved into the memory so the robot can further optimize its path in successive travels.

Here is an example of how the algorithm operates in a completely unknown environment. Initially, an empty map is generated in the memory of the robot. The start and finish locations are plotted and all moves are given the same cost. Therefore, the algorithm chooses an optimal path as a direct line from the start to the finish points. The robot then begins its journey. As the robot performs each move, the blank map is checked against the surroundings. If any difference is found, these are noted on the map and a new cost is associated with that location. If a newly found obstacle lies on the planned path, the algorithm will replan the path locally based on the new cost information. The robot will then move along its new path until either more obstacles or the goal is located. Figure 2 shows the D* algorithm at work (refs. 6 and 7).

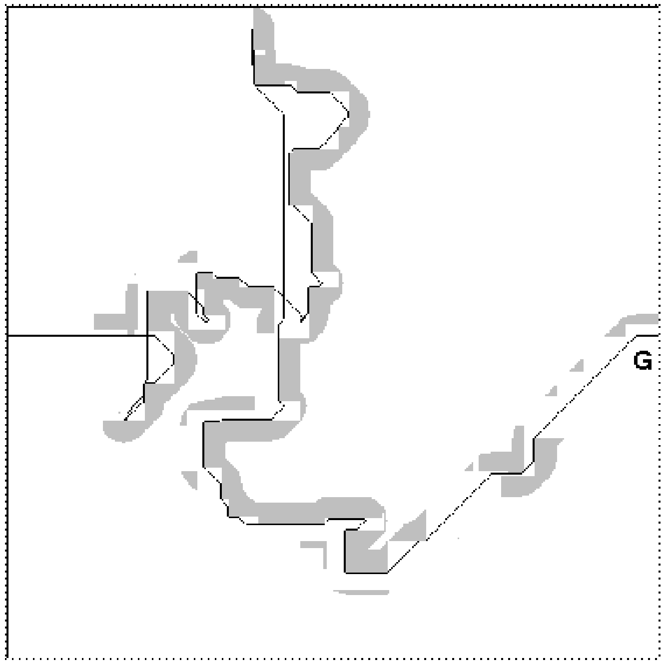
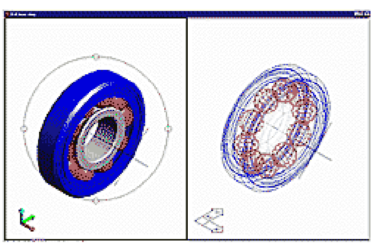


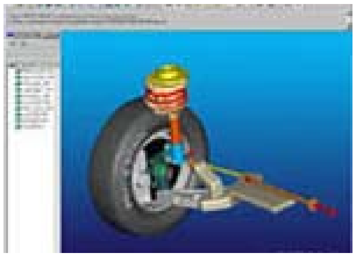
Figure 2
Path generated using D*

Computer Aided Design Programs

AutoCAD

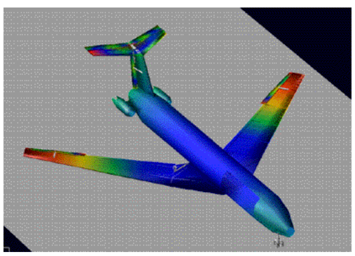
 AutoCAD, created by Autodesk, is a useful computer-drafting tool. It allows for very detailed drawings. Its drawing format is compatible with most other drawing and simulation programs. The latest releases of AutoCAD include extensive 3D packages to create models, extending the software far beyond a drafting tool. AutoCAD takes the more traditional view of drafting; almost all components of a design are drawn line by line, then the lines are connected as a rigid three-dimensional object (ref. 8).

Pro-Engineer



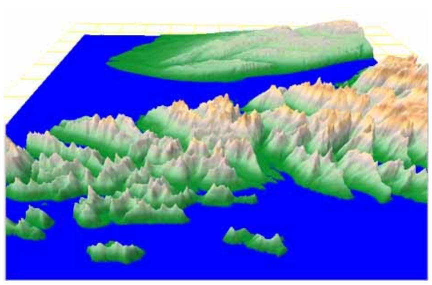
Pro-Engineer, created by the Parametric Technology Corporation, is an extremely powerful software package. Pro-E is very versatile. It has been oriented for 3D design since its inception and it already includes general motion simulation as a feature of the program. Designs can be tested without leaving the Pro-E environment. This allows for design that is more rapid and modification that is much quicker (ref. 9).

ADAMS



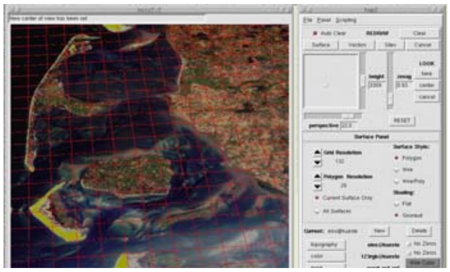
ADAMS is a series of programs developed by Mechanical Dynamics and is commonly used for general-purpose simulation. ADAMS can be used to simulate almost any type of vehicle on land or in the air and is very good at modeling most structures. ADAMS is used to model the motion of components within a vehicle, and is not useful for modeling complex terrain for the purposes of navigation simulation (ref. 10).

MapRender 3D



MapRender 3D is an advanced software package to create very detailed relief maps of the earth. The software contains detailed databases on all areas of the world. It is capable of creating both two and three-dimensional views. The software may be able to be adapted for vehicle surfaces (ref. 11).

GRASS (Geographic Resources Analysis Support System)



GRASS was originally produced by the U.S. Army Construction Engineering Research Laboratories (USA-CERL) branch of the US Army Corp of Engineers. It is used for data management, image processing, graphics production, spatial modeling, and visualization of many types of data, and as a tool for land management and environmental planning by the military. It is not very useful as a terrain editor for use in simulation of a mobile robot crossing a complex terrain (ref. 12)

Projects and Research

IS Robotics

The Gecko. The Gecko is a Defense Advance Research Projects Agency (DARPA) funded project with the intent of creating a small mobile robot with the ability to attach itself and move over non-smooth, non-horizontal surfaces. The project actually falls under the name of the "Component Technologies for Climbing" program, but it takes its name and inspiration from the Gecko lizard and efforts at the Poly-PEDAL Lab at the University of California at Berkeley. Current research includes learning techniques of adhesion that a gecko uses to cling to any surface and transition from surface to surface no matter what the difference in characteristics or orientation of the two surfaces may be. They are working with the concept that the microscopic hairs on a gecko's feet (known as setae) have an intermolecular attraction (called Van der Waals forces) with the surface they are in contact with. IS Robotics is hoping to be able to duplicate this adhesion and incorporate it into a very small self-contained walking robot. IS Robotics has been working with the Poly-PEDAL Lab at University of California at Berkeley to investigate some of the various climbing mechanisms found in nature. They are also looking at the Gecko's spine flexibility to aid in surface transitioning for the robot.

The developers already realize that, for a robot application, it may not be feasible to use a single form of attachment for all surfaces since they have been unable to duplicate the Van der Waals forces. They may find the use of claws for softer surfaces and sticky adhesives for smoother, hard surfaces more feasible and cost effective.

The next step for IS Robotics, once the adhesion technique is developed, is to produce a small autonomous mobile micro-robot that can move in a three dimensional unstructured environment. The current work is only one important step in the whole process for IS Robotics. Once the adhesion technique is perfected, work will begin to allow the robot to autonomously navigate through the difficult terrain and map its progress and surroundings. The staff at IS Robotics hopes to develop algorithms that go far beyond the path planning algorithms of today's autonomous robots (refs. 13 and 14).

Ariel. IS Robotics, along with help from the Poly-PEDAL Lab at the University of California at Berkeley and funding from DARPA, has developed an extremely resourceful amphibious robot named Ariel (fig. 3). Ariel is an autonomous hexapod robot modeled after a crab. They are used in surf zones to seek out and destroy mines and other shoreline obstacles. Several of the Ariel robots can be deployed at once and collectively search for the explosives to be removed. Once a target is found, Ariel will do one of two things: it will either attach itself to each device and await a detonation signal, or it will deploy an explosive payload and move to a safer area. Once the signal is sent, all Ariel robots detonate their acquired targets.

Figure 3
Ariel robot

Ariel is an extremely stable robot. It is capable of changing its angle of attack quickly for climbing and burrowing, as well as aiding it to remain stable in shifting water currents. Ariel is also capable of operating at any position, including completely inverted. Each leg is at the side of the body and has two degrees of freedom to allow Ariel to walk while standing on its head. Ariel shows the capabilities of small legged robots in harsh environments and provides some insight into the climbing ability of legged robots. Most importantly, the autonomous nature of Ariel provides insight into goal recognition and payload deployment for an autonomous robot in an unknown environment (refs. 13 and 14).

University of Portsmouth, UK

Robug IIs. The Robug IIs was the first robot built by the University of Portsmouth with the ability to climb walls. The robot is quite large at a length of about 1.5 m and has the appearance of a giant spider with its ‘knees’ high above its body to help give its stability. It is capable of climbing vertically up smooth surfaces and is large enough to scale small ledges and stairs. It can also carry a weight equivalent of approximately 80 lbs. up a smooth wall. It is a completely tele-operated design, which means that external power and control is required. It is not autonomous and is not highly mobile. However, the project gives some insight towards surface adhesion.

Robug III. Robug III (fig. 4) is the latest concept design at the University of Portsmouth; it is the next generation Robug. It will be slightly smaller than Robug IIs but will still be quite large for a climbing robot. It will have eight four-jointed legs and be capable of climbing vertical surfaces. It will also be tele-operated, but will include more on-board sensing features and better surface adhesion than Robug IIs. Interest in the Robug project continues only because of the work done in wall climbing and surface adhesion (ref. 15).

Figure 4
Artist view of Robug III

MIT Leg Laboratory (part of The Artificial Intelligence Lab)

The focus of work at the MIT Leg Laboratory is to perform experiments on active balance in dynamic legged locomotion. The lab builds and tests many designs for walking robots that mimic creatures such as kangaroos, birds, reptiles, and ants. They are developing control algorithms for the single and multi-legged robots (fig. 5). They are showing excellent control with hopping, walking, and running robots and are showing remarkable stability in recovery from small disturbances. Most of the robots are semi-autonomous.

The MIT Leg Lab is developing simulation programs to fine-tune the behavioral control of their robots. They recreate the robot in the computer environment and alter the behavioral algorithms for quick optimization. They are working with an automated way of retuning control algorithms to allow fast adaptation between designs and are even going as far as simulating robots with no sensing or reaction systems to watch how they move around a terrain. Their work can be extremely useful for balance and behavioral control of future robots (ref. 16).

Figure 5
Troody, one of MIT's walking biomimetic robots

Robotics Institute at Carnegie Mellon University

Millibots. The Millibots (fig. 6) are a colony of small autonomous robots. They work as a group to accomplish their tasks, as ants do in a colony. They are deployed as a group and communicate among one another in order to complete their assignments. Each Millibot contains a sensing platform and uses a radio frequency (RF) link transceiver to communicate amongst each another. Millibots also carry a modular payload that they use at their specified goal to complete their mission.

Although the robots are limited to rather smooth terrain due to their small wheels as the only means of locomotion, this project shows great potential in other areas. Carnegie Mellon is showing how several inexpensive autonomous robots can interact to complete their tasks faster and are experimenting with payload deployment techniques (ref.17).

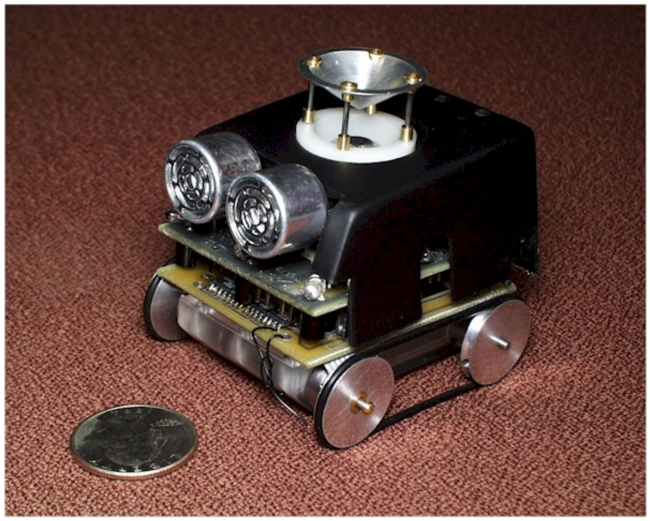


Figure 6
Millibot shown with a quarter

Real and Virtual Environment for Multiple Robots (CyberRAVE). CyberRAVE is a general-purpose framework to run and simulate multiple robot systems (fig. 7). It allows real and virtual robots to interact in a virtual environment. The primary function of the software is to aid designers in programming their robot. A simulated version of the design is first programmed with the desired behaviors. Then the control program is transferred to the real robot. Virtual sensors can also be placed onto a real robot so that it can interact with its virtual counterparts. Writing and optimizing the code in the virtual environment greatly reduces the time and effort spent programming a robot (ref. 17).

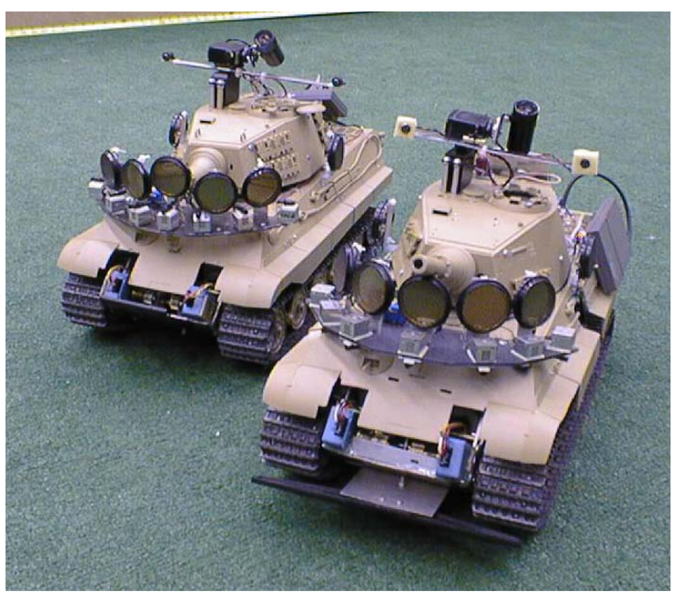


Figure 7
CyberRAVE robots used to interact with software

Dynamic Mission Planning for Multiple Robots. Work for this project is based on the D* algorithm developed at the Robotics Institute at Carnegie Mellon University. The algorithm was adapted to be able to handle multiple robots and multiple goals. The robots work as a team to reach all of the goals and return to their base in the shortest amount of time. Each robot is remotely linked to a mission planner located at the base so that the robots may communicate amongst one another. The mission planner is the central brain of the team; all received information passes to it and all commands are sent from it.

The experiments conducted at the Robotics Institute include using two robots with four goals and three robots with six goals. Note that each robot does not need to reach each goal; it is only required that each goal be reached once. In each experiment, the mission planner knows only minimal information about the structure of the environment. Using this information, along with the location of the base and the goal locations, the mission planner assigns an optimal path to the robots to reach the goals and return to the base in the shortest total time. The robots then begin their travel. As each robot discovers new information about the environment, the information is relayed back to the mission planner and it is added to the map.

Each robot has the ability to avoid the newly detected obstacles without intervention from the mission planner. It will locally adjust its path to its assigned goal and update its position to the mission planner. Meanwhile, the mission planner is continually calculating if the current chosen paths are optimal for the team. If a decision is made that it may be faster for robots to swap goals or change the order they reach the goals, the mission planner will instruct the robots of the change and plan new paths for them. This is known as dynamic mission planning.

The scenario with three robots and six goals was simulated 1,000 times using two methods: one static where the order of the goals do not change and one dynamic, where the order of the goals can change. The dynamic mission planning resulted in a 25% higher efficiency than the static planning. With dynamic planning, the missions of the robots were changed on average 5.2 times per run (ref. 18).

Bridge Inspection with Serpentine Robots. The Robotics Lab Department of Carnegie Mellon University is currently working on a serpentine robot for the purpose of bridge inspection. Bridge inspections are costly, with rigging and traffic control consuming over 40 to 50% of the cost. The possibility of creating a flexible serpentine robot capable of reaching many difficult places on these bridges may prove to be cost-efficient, safer, and improve the amount of information that can be gathered, while decreasing traffic delays caused by normal inspections. Conventional mobile robots cannot perform these bridge inspections due to the lack of their flexibility to reach all locations. This serpentine robot will possess multiple joints, giving it superior flexible ability to reach any points of high difficulties. Building and programming this serpentine robot will be difficult. A big problem is the control of the robot because the planner must account for all the joints and the possible degrees of freedom exercised by the mechanism. The robot will be able to use a roadmap or a geometrical structural plan of the bridge it is inspecting as a robotic planning field. This roadmap, which may be retrieved from a CAD model of the bridge, will enhance the vision of the serpentine robot, allowing it to plan for paths that will guarantee its sensors will “see” all locations of the bridge. However, if no roadmap can be provided, the serpentine robot can construct the roadmap of the bridge as it is inspecting using the gathered data from its sensors.

This project will prove extremely useful in the research in creating better autonomous robot. If programmed and built correctly, this autonomous robot may be capable of reaching hard to reach places that other conventional robots cannot. Unfortunately, the power consumption it needs is extremely high, needing an external power source. A better power unit must be researched in order to allow it to work with more efficiency (ref. 17).

Robot Coverage for De-mining. The Robotics Institute of Carnegie Mellon University is in the process of developing an autonomous robot for land mine detection. De-mining an area is a dangerous and extremely costly task. To thoroughly de-mine an area, the robot must pass the mine-detection sensor over all the possible points that might hide a mine without setting them off. Therefore, a complex path planning system must be used to traverse through such a complex and dangerous region.

This complex path planning system will greatly contribute in the understanding of analyzing the means of traversing through a hazardous terrain. The group is also attempting to meld a probabilistic planner technology that can significantly increase the capabilities of the current sensors for faster and more efficient results (ref. 17).

NASA Ames Research Center

Serpentine Robotics Project. NASA's serpentine robots (fig. 8) are comprised of low degree of freedom modules linked together to form a highly flexible robot. The modules work together to create snake-like movements and allow for high mobility in complex terrain. The robot has been labeled as "hyper-redundant," which means that it is very robust and can continue to function properly if a few of the modules stop working.

NASA is trying to study the mobility of serpentine robots to make them more efficient. They are already showing that the serpentine robot can easily overcome obstacles much larger than itself; and it can maneuver in areas much more complex than traditional wheeled and legged robots. In zero gravity environments, one end of the snake can be fixed to a surface and it can be used as a very articulate arm to assemble components in space. Work also shows how many low-intelligent robots can be linked together to be much more efficient than a single intelligent robot.

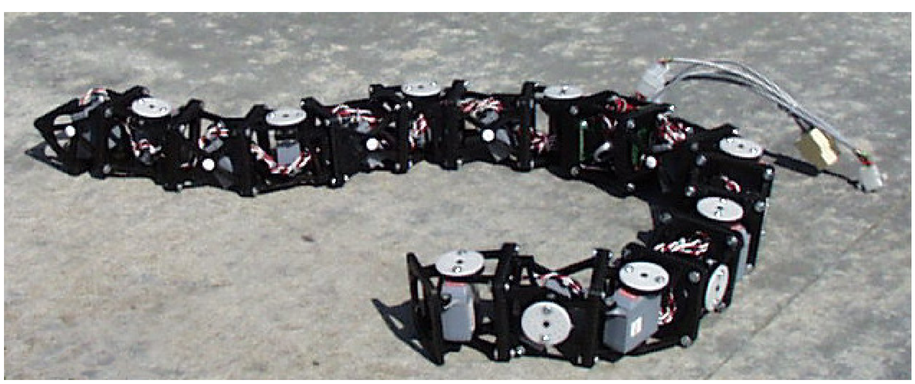


Figure 8
NASA's serpentine robot

The serpentine robots at NASA show great potential in design and programming for autonomous navigation of a micro-robot through a complex 3D environment. Its stable and adaptable shape provides an excellent platform for maneuvering through difficult terrain. The main drawback to the design is that it is very power consuming, difficult to program, and currently must be tele-operated. The robot is currently a bit longer than most micro-robots, but may be able to be shortened by removing some links in the future (ref. 1)

University of Minnesota

The Loon. A high-speed autonomous wheel type robot built out of LEGOs by a group of students from University of Minnesota (fig. 9), consisting of a large robot and a smaller counterpart, the baby Loon, riding on the back of the larger one. The Loon robot was built for a "Schedule a Meeting" competition. Both robots are equipped with a ring of five Polaroid ultrasonic sensors and a single Polaroid sensor on a 360-deg freedom turret to detect objects and obstacles. Four close range Infrared detectors placed on the front, back, and each side of the Loon robots allow them to sense heat signatures, in order to detect and avoid human presence. Shaft encoder on one of its wheels for position wheel motion (PWM) calculations and dead reckoning capabilities were added. Even though the Loon robot is programmed with an internal map, it is capable of mapping another path in case an obstacle is present in the original path. When needed at two different locations, the Loon robot is capable of transferring necessary data to the baby Loon. It will then release its smaller counterpart so that the baby Loon may proceed to one of its destinations.

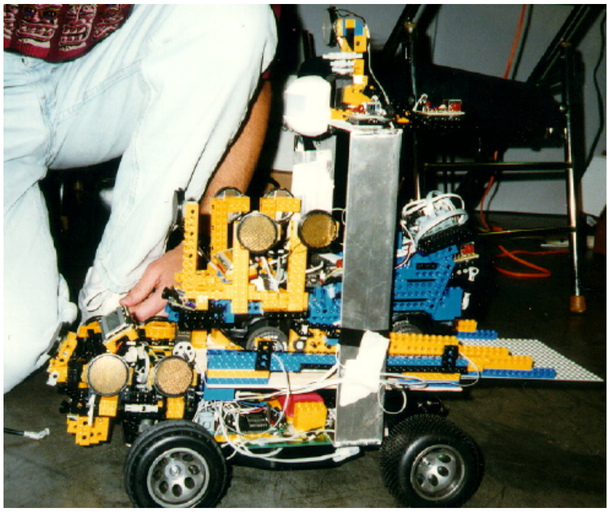


Figure 9
Mama Loon with Baby Loon on back

The Loon project is an interesting project. It has a good path planning system, allowing the robot to recalculate its path if the original path is not possible. The most noteworthy aspect is that the Loon is able to release a smaller version of itself, completely autonomous once launched. The Loon is composed of LEGOs; therefore it cannot sustain a lot of physical damage without breaking down. This is understandable because LEGOs are more efficient in the prototyping and construction stage (ref. 19).

Walleye. The Walleye (fig. 10) is one of the many autonomous robots built by the engineering lab from the University of Minnesota. It uses three 6,811 microprocessors linked together through synchronous serial ports, one controlling the gripper, one for decision making, and the third one for vision. Walleye is capable of visually finding Styrofoam cups and empty cans, pick them up and dispose them in a trash or recycling bin, using a small black and white camera (160 by 160 pixels).

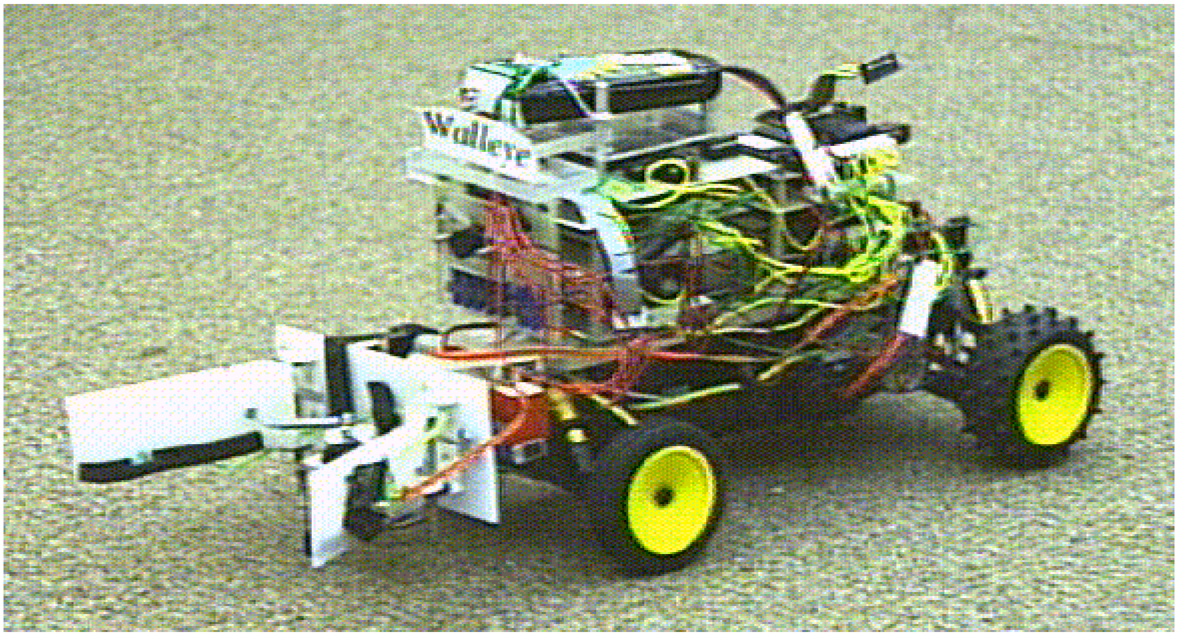


Figure 10
The Walleye

Visual recognition capability makes Walleye a very useful autonomous robot. Slight improvements in the system would make this technology extremely useful. These improvements may be using a color camera to distinguish between colors and recognizing more difficult patterns or objects (ref. 19).

Stanford University

Motion Planning for a Team of Mobile Robots. The Computer Science Department at Stanford University has been working on algorithms to find optimal paths for robots in a defined workspace for the purposes of detecting and tracking another moving object in that space. The robots used to track are known as “pursuers” and the robots being tracked are known as “evaders”. The team of pursuers can perform three separate operations:

- Cover a region to guarantee the target will be found
- Follow a target without losing line of sight
- Cover a region to create a three-dimensional map for other robots to use in path planning processes

This work shows great insight into using a team of robots to complete a mission. Robots can be deployed in different places to cover the area quickly. A complex image map can be created of the surface that can be shared with the others in the team. Then the group can seek out its targets in a much more efficient manner than a single robot searching in an entirely unknown environment (refs. 20 through 22).

U.S. Army Research Lab

DEMO III Unmanned Ground Vehicle. The Demo III program is the latest of the Department of Defense's efforts into the area of unmanned autonomous ground vehicles (UGV). It is a medium-sized, four-wheel vehicle capable of travelling though a terrain consisting of hills, valleys, and obstacles to avoid. Currently, the Demo III uses an internal map and is given an initial trajectory to follow. An operator in a second vehicle gives the Demo III travel orders then follows closely behind to observe the actions of the UGV. The vehicle can detect obstacles not on its map, chart them, then plot a new course to avoid the new obstacle to reach its goal. The Demo III can also “decide” whether or not to travel over or around different size hills. Currently it has trouble with negative slopes, or dips in the road. It may be confused by small ditches. The Demo III can travel off road, but experiences navigation trouble in heavily wooded areas. This is due to the vision system being unable to comprehend the many trees and leaves that may be swaying in the wind.

The current version incorporates three safety mechanisms to keep the vehicle under human control. First, emergency buttons on the vehicle can cut off the power supply. Second, a remote transmitter can be used to cut the power. Third, if the following vehicle falls too far behind the Demo III the power will be cut off.

The Demo III is slated to be primarily used as aid to infantry and as a mine sweeper and enemy detection device. However, the work in autonomous navigation may be very useful to be adapted to more complex environments (ref. 23).

DISCUSSION

Tremendous effort by companies and universities in constructing autonomous robots demonstrate enormous potential for further advancement. With improvements in several key areas in robotics and electronic technology, a fully autonomous robot capable of searching for goals in a complex three-dimensional environment may be achievable in the near future.

Power units, or batteries, that are readily available are very bulky, heavy, and do not provide enough energy to sustain an autonomous robot for a long period of time, making them very difficult to manage. The size, weight and power these energy units provide are huge factors and a smaller, more efficient source of power may be preferable. A more efficient power supply will allow a robot to operate for a longer period of time, creating a better opportunity to complete its objectives. A serpentine or legged robot would benefit radically due to their high consumption of energy for movement. A small and efficient power source would allow them to move freely without being attached to an external energy supply.

Locomotion for a robot travelling in complex terrain may not be limited to one form. A hybrid of the three basic forms may prove more beneficial than the use of only one. A robot may use wheels to travel smooth surfaces, inclines, and declines. But it may also use serpentine motion to bridge

gaps or use legs like hooks to climb. Recommendations from various professors suggest that the best form of locomotion to use at this time is a wheel-type transportation system. The energy consumption is considerably lower than the other options and its reach is severely underestimated. Coupled with a type of hook line, a wheel-type robot will be able to scale a relatively larger obstacle. However, if a serpentine robot can be fully realized as a self-contained, autonomous robot, it could provide an excellent platform on which to begin construction of a robot with possibly no limitations in mobility.

Path planning algorithms must be greatly enhanced so they may plan efficiently in a complex three-dimensional environment. The robot must understand its position in three dimensional space. Even though it may find itself in the same planar co-ordinates it did previously, it may now be at a new elevation. A situation like this could prove confusing and could create errors for current processes as the robot may think it has returned to an already mapped position but at the same time will not detect any other previously found features. Mapping in three dimensions is also important so that a robot can find its position more accurately. Moving two feet on level ground will put the robot in a much different location than moving 2 ft at a 30-deg slope. Sensors are important to robot positioning. Even the best of GPS, which can find position within one millimeter, is not good enough. It only finds that position in two dimensions, it takes no account for elevation. Therefore, even high performance GPS cannot be used as the only positioning system. A localized system such as dead reckoning or the Voronoi system would prove more beneficial in conjunction with sensors to aid in calculations for robot angle and elevation.

Sensors are also very important to robot functions. A robot requires sensors to move, detect its surroundings, and to help decide what moves to make next. An environment loaded with obstacles may need more complex vision systems, or ways of “seeing” the obstacles. The use of visible light and other light cameras may allow a robot to comprehend its environment better versus using touch and proximity sensors to just find the locations of potential obstacles. The camera system may even be used in conjunction with a detailed layout of the vehicle it is on. It may be able to compare what it sees with the map and be able to locate itself on that map, reducing the time needed to explore the surface and allowing the robot to find its targets much more quickly (ref. 24). Two major concerns exist for vision systems and sensors in general. The first concern is the size of these cameras; they may be much too large to mount on a micro-robot. The second concern is the possibility of harsh weather conditions. Visible light cameras may be impaired in darkness or fog, and other vision sensors may be impaired in rain.

The material used in building these robots must be carefully chosen for maximum durability and efficiency. Harsh weather conditions may rust metallic surfaces and water may affect a robot’s operation. The entire robot must be thoroughly weatherproofed. Another problem commonly found in robot construction is finding small actuators that are strong enough to perform the motions of the autonomous robots, particularly with robots that move through leg or serpentine locomotion.
The integration of robot design software with simulation software and terrain mapping software is essential to the future of robotics development. By using a single set of software that is capable of designing the physical appearance of the robot as well as code its behavior, testing its functionality, and creating the complex environment, the amount of time and money spent in developing such an advanced autonomous robot would be reduced significantly.

CONCLUSIONS

A study was conducted to look into the maturity of the fields of robotic technology. The information was compiled to locate applicable technology for further study to help build a small, autonomous mobile robot that is capable of moving around a complex, three-dimensional environment with the purposes of inspection and goal locating. The technical difficulties and concerns involved with developing such a powerful robot suggest that a complete final testable prototype is still several years beyond the present. Specific projects were introduced, along with the advantages and disadvantages of each project. Recommendations for increased research and advancement have been made to help realize the ultimate goal of creating a small autonomous robot with the previously mentioned capabilities.

REFERENCES

1. Nova Laboratory, Inc., www.novalab.com.

2. Choset, Howie; Nagatani, Keiji; and Rizzi, Alfred, "Sensor Based Planning: Using a Honing Strategy and Local Map Method to Implement the Generalized Voronoi Graph," Mechanical Engineering and Robotics, Carnegie Mellon University, Pittsburgh, PA, 1997.

3. Nagatani, Keiji; Choset, Howie; and Thrun, Sebastian, "Towards Exact Localization without Explicit Localization with the Generalized Voronoi Graph," Mechanical Engineering and Robotics, Carnegie Mellon University, Pittsburgh, PA, 1998.

4. Auyeung, Tak, PhD, "Popular Micromouse Algorithms, Part IV: A* Algorithm," Robot Science and Technology Magazine, April 1999.

5. Stanford University, www.db.stanford.edu/~burback/watersluice/node84.html.

6. Stentz, Anthony, "Optimal and Efficient Path Planning for Partially-Known Environments," The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 1994.

7. Stentz, Anthony, "Map-Based Strategies for Robot Navigation in Unknown Environments," The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 1996.

8. Autodesk, www.autodesk.com.

9. Parametric Technology Corporation, www.ptc.com.

10. Mechanical Dynamics, www.adams.com.

11. Digital Wisdom, maprender3d.com.

12. USA-CERL, www.baylor.edu/~grass.

13. IS Robotics, www.isr.com.

14. Poly-PEDAL Lab at U. C. Berkeley, polypedal.berkeley.edu/Bioinspire/Robotics.html.

15. University of Portsmouth, www2.ee.port.ac.uk/~robotwww/mech.html.

16. MIT Leg Lab, www.ai.mit.edu/projects/leglab.

17. Robotics Institute at Carnegie Mellon, www.ri.cmu.edu.

18. Brumitt, Barry L. and Stentz, Anthony, "Dynamic Mission Planning for Multiple Robots," The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 1996.

19. University of Minnesota, www.cs.umn.edu/Research/airvl/minirob.html.

20. LaValle, Steven M.; Gonzalez-Banos, Hector H.; Becker, Craig; and Latombe, and Jean-Claude, “Motion Strategies for Maintaining Visibility of a Moving Target,” Computer Science Department, Stanford University, Stanford, CA, 1997.

21. LaValle, Steven M.; Lin, David; Guibas, Leonidas J.; Latombe, Jean-Claude; and Motwani, Rajeev, “Finding an Unpredictable Target in a Workspace with Obstacles,” Computer Science Department, Stanford University, Stanford, CA, 1997.

22. Gonzalez-Banos, Hector and Latombe, Jean-Claude, “Planning Robot Motions for Range-Image Acquisition and Automatic 3D Model Construction,” Computer Science Department, Stanford University, Stanford, CA, 1998.

23. Bornstein, Jon and Shoemaker, Chuck, “The Demo III Program: A Testbed for Autonomous Navigation Research and Experimentation,” Weapons and Materials Research Directorate, U.S. Army Research Laboratory, October 1999.

24. Kosaka, Akio and Pan, Juiyao, “Purdue Experiments in Model-Based Vision for Hallway Navigation,” Robot Vision Laboratory, Purdue University, West Lafayette, IN, 1995.

DISTRIBUTION LIST

Commander
Armament Research, Development and Engineering Center
U.S. Army Tank-automotive and Armaments Command
ATTN: AMSTA-AR-WEL-T (2)
AMSTA-AR-GCL
AMSTA-AR-FSF (??)
Picatinny Arsenal, NJ 07806-5000

Defense Technical Information Center (DTIC)
ATTN: Accessions Division (12)
8725 John J. Kingman Road, Ste 0944
Fort Belvoir, VA 22060-6218

Director
U.S. Army Materiel Systems Analysis Activity
ATTN: AMXSY-EI
392 Hopkins Road
Aberdeen Proving Ground, MD 21005-5071

Commander
Chemical/Biological Defense Agency
U.S. Army Armament, Munitions and Chemical Command
ATTN: AMSCB-CII, Library
Aberdeen Proving Ground, MD 21010-5423

Director
U.S. Army Edgewood Research, Development and Engineering Center
ATTN: SCBRD-RTB (Aerodynamics Technology Team)
Aberdeen Proving Ground, MD 21010-5423

Director
U.S. Army Research Laboratory
ATTN: AMSRL-OP-CI-B, Technical Library
Aberdeen Proving Ground, MD 21005-5066

Chief
Benet Weapons Laboratory, CCAC
Armament Research, Development and Engineering Center
U.S. Army Tank-automotive and Armaments Command
ATTN: AMSTA-AR-CCB-TL
Watervliet, NY 12189-5000

Director
U.S. Army TRADOC Analysis Command-WSMR
ATTN: ATRC-WSS-R
White Sands Missile Range, NM 88002

Commander
Naval Air Warfare Center Weapons Division
1 Administration Circle
ATTN: Code 473C1D, Carolyn Dettling (2)
China Lake, CA 93555-6001

GIDEP Operations Center
P.O. Box 8000
Corona, CA 91718-8000